REMARKS

Claims 1-7, 11, 12, 13, 15-18, 27-29 and 31-33 are pending in the application. Claims 8-10, 14, 19-26, 30 and 34 are canceled. Claims 1-7, 12, 13, 16-18, 27 and 31 are currently amended. No new matter has been introduced.

Objections to the Drawings

The Examiner objected to the drawings under 35 CFR Section 1.83(a) for failing to show the limitations of claims 24-34. The Examiner's objections are respectfully traversed.

With regard to claims 24-26, the Examiner does not set forth a specific basis for the objection. Based on the Examiner's rejection of claims 1 and 24-26 as not enabled and as directed to non-statutory subject matter (which are addressed and traversed below), the Examiner's objection appears to be that specific embodiments of using the method of claim 1 in (i) spectral analysis, (ii) filtering in the frequency domain and (iii) performing polyphase transformations are not separately illustrated. While the Examiner's contention that the specification did not enable use of the method of claim 1 in connection with spectral analysis, filtering in the frequency domain and polyphase transformations is respectfully traversed, the current claims do not recite these limitations. It is noted that one of skill in the art after reviewing the specification would have known how to use the method of claim 1 in connection with spectral analysis, digital signal filtering and polyphase transformations. See MPEP 608.02. It is further noted that the setting in which an invention will be used does not require specifics in the specification, but may be mentioned in general terms. See MPEP Section 608.01 and MPEP Section 2164.01(a). It is also respectfully submitted that additional drawings showing the use of, for example, the method of claim 1, in filtering in the frequency domain would be neither necessary nor helpful to understanding the invention. This also appears to address at least part of the Examiner's objections with regard to claims 30 and 34. Nevertheless, to expedite prosecution claims 24-26, 30 and 34 are canceled.

With regard to claims 27-29 (claim 30 is canceled), the Examiner objected to the drawings as allegedly not showing "computing remaining butterfly stages of the N-point FFT/IFFT with a single iterative loop wherein each butterfly operation in each stage of a loop

iteration is computed on a single respective processor in the plurality of processors and there is no data dependency between butterflies in a stage of an iteration of the loop." With regard to claims 31-33 (claim 34 is canceled), the Examiner objected to the drawings as allegedly not showing "compute remaining butterfly stages of the N-point FFT/IFFT with a single iterative loop wherein each butterfly operation in each stage of a loop iteration is computed on a single respective processor in the plurality of processors and there is no data dependency between butterflies in a stage of an iteration of the loop." While the Examiner's objection is respectfully traversed, the claims as amended do not recite the language to which the Examiner points. As discussed in more detail below, it is believed that the drawings sufficiently illustrate the claimed subject matter.

The Claims Are Enabled by the Specification

The Examiner rejected claims 24-34 under 35 USC Section 112, first paragraph, as not enabled. The Examiner's rejections are respectfully traversed. While the Examiner's rejection is traversed, the new claims do not recite the language to which the Examiner contends is not enabled, which renders this basis for rejection moot. It is noted that claims 24-26, 30 and 34 are canceled. It is believed that claims 27-29 and 31-33 as amended are enabled by the specification as filed.

There is a strong presumption that the specification contains an adequate disclosure as filed and the Examiner has the burden of presenting evidence or reasons why one of skill in the art would not recognize that the written description provides support for the claims. MPEP 2163(II). Further, there is no *in haec verba* requirement. MPEP 2163. Rejection of an original claim as not supported by the specification should be rare. *Id.* For computer related claims, a disclosure of a processor capable of performing a function and the function is generally sufficient to enable the claims. *See, e.g., Fonar Corp. v. General Electric Corp.*, 107 F.3d 1543, 1549 (Fed. Cir. 1997) ("As a general rule, where software constitutes part of a best mode of carrying out an invention, description of such a best mode is satisfied by a disclosure of the function of the software. This is because, normally, writing code for such software is within the skill of the art, not requiring undue experimentation, once its functions have been disclosed. ...

Thus, flow charts or source code listings are not a requirement for adequately disclosing the functions of software."); *In re Hayes Microcomputer Prods. Lit.*, 982 F.2d 1527, 1534-35 (Fed. Cir. 1992) (One skilled in the art would know how to program a microprocessor to perform the necessary steps described in the specification. Thus, an inventor is not required to describe every detail of his invention. An applicant's disclosure obligation varies according to the art to which the invention pertains. Disclosing a microprocessor capable of performing certain functions is sufficient to satisfy the requirement of section 112, first paragraph, when one of skill in the relevant art would understand what is intended and know how to carry it out.").

Claims 5-6 and 16-20 Are Sufficiently Definite

The Examiner rejected claims 5-6 and 16-20 under 35 USC Section 112, second paragraph, as indefinite. The Examiner's rejections are respectfully traversed. Specifically, the Examiner contends one of skill in the art would not know what was meant by "computer-readable memory medium." Applicants disagree that one of skill in the art would not have known what was meant by "computer-readable memory medium," and as noted above there is no *in haec verba* requirement. Further, the Examiner had no trouble identifying examples of what was meant by the phrase. Nevertheless, claims 5, 6 and 16-18 have been amended to recite "computer-readable storage medium" to expedite prosecution. Support for this amendment can be found in the application as filed on page 5, lines 17-21. It is believed that this amendment obviates any possible concerns regarding the definiteness of claims 5-6 and 16-18. Claims 19 and 20 have been canceled.

The Claims Are Directed to Statutory Subject Matter

The Examiner rejected claims 1-4, 7, 10-15 and 24-34 under 35 U.S.C. § 101 as allegedly being directed towards non-statutory subject matter. The Examiner's rejections are respectfully traversed. Claims 10, 14, 24-26, 30 and 34 are canceled, which renders this basis for rejection moot with respect to those claims. The other claims rejected as non-statutory are addressed below.

As an initial matter, it is noted that during prosecution much of the Examiner's concerns centered on whether the claims were directed to statutory subject matter. The state of the law regarding whether computer-related claims are directed to statutory subject matter is an issue that has been in flux. Recently, and subsequent to the Examiner's Final Office Action, new Interim Patent Subject Matter Eligibility Examination Instructions (the "Interim Instructions") were issued by the Patent Office, which set forth example formats for claiming that Applicants believe would, if adopted by the Applicant, address the Examiner's concerns regarding statutory subject matter and expedite prosecution of this case. While it is believed the claims prior to amendment were directed to statutory subject matter, to expedite prosecution minor amendments have been made to independent claims 1, 5, 27 and 31 to put them into formats that more closely resemble the formats of the statutory examples contained in the Interim Instructions.

The Interim Instructions contrast five hypothetical computer-related claims, claims 2-6. Hypothetical claims 4 and 6 are not statutory, but hypothetical claims 2, 3 and 5 are statutory. The difference between non-statutory hypothetical claims on the one hand and the statutory hypothetical claims on the other hand is that the statutory claims recite a machine in connection with a part of the claimed embodiment of the hypothetical claim that is more than mere insignificant extra-solution activity. For example, under the interim guidelines hypothetical claim 4 is not statutory, but hypothetical claim 5 is statutory because under the broadest reasonable interpretation of hypothetical claim 5, the step performed using a microprocessor requires a particular programmed processor and the step is not mere insignificant extra-solution activity. Hypothetical claim 6 recites that the search results are electronically downloaded, but is otherwise identical to hypothetical claim 4. Hypothetical claim 6 is non-statutory because the only step requiring a machine is insignificant extra-solution activity.

Turning to the language of the claims at issue here, independent claim 1 as amended recites, "the method comprising using a multiprocessor computing system having a plurality of processors P to perform the steps of: [all of the claimed steps]." Claim 1 is directed to statutory subject matter for the same reasons that hypothetical claim 5 of the Interim Instructions was directed to statutory subject matter. Under the broadest reasonable interpretation, the multiprocessor computing system must be configured or programmed in a

particular manner to perform ***all*** of the recited method steps. Thus, a particular machine is recited, the multiprocessor computing system. In addition the machine imposes a meaningful limitation and is more that an insignificant extra-solution activity because ***all*** of the method steps are performed by the claimed multiprocessor computing system, and thus at least one of the steps must be more than mere insignificant extra-solution activity. To the extent the Examiner contends otherwise, it is respectfully submitted that the Examiner is instead raising a rejection under Section 102 or 103, for which the Examiner is required to establish a *prima facie* case of anticipation or obviousness by pointing to prior art. Thus claim 1 is directed to statutory subject matter. Claims 2, 7 and 11 depend from claim 1 and are directed to statutory subject matter at least by virtue of their dependencies.

Independent claim 3 is a means-plus-function product claim. As amended claim 3 recites, "the multiprocessor system having a plurality of processors P and configured to implement: [all of the means-plus-function elements]." Claim 3 is directed to statutory subject matter because it recites a particular machine, the multiprocessor system, configured to implement all of the means-plus-function elements. Under the broadest reasonable interpretation, the multiprocessor system must be configured or programmed in a particular manner to implement ***all*** of the recited means-plus-function elements. Where all of the means-plus-function elements are implemented by the claimed multiprocessor system, at least one of the elements must be more than mere insignificant extra-solution activity. To the extent the Examiner contends otherwise, it is respectfully submitted that the Examiner is instead raising a rejection under Section 102 or 103, for which the Examiner is required to establish a *prima facie* case of anticipation or obviousness by pointing to prior art. Thus, claim 3 is directed to statutory subject matter. Claims 4 and 12, 13 and 15 are directed to statutory subject matter at least by virtue of their dependencies.

Independent claim 27, as amended, recites "using a multiprocessor computing system having a plurality P of processors to: [all of the recited method steps.]" Under the broadest reasonable interpretation, the multiprocessor computing system must be configured or programmed in a particular manner to perform ***all*** of the recited method steps. Thus, a particular machine is recited, the multiprocessor computing system. In addition the machine imposes a

meaningful limitation and is more that an insignificant extra-solution activity because ***all*** of the method steps are performed by the claimed multiprocessor computing system, and thus at least one of the steps must be more than mere insignificant extra-solution activity. To the extent the Examiner contends otherwise, it is respectfully submitted that the Examiner is instead raising a rejection under Section 102 or 103, for which the Examiner is required to establish a *prima facie* case of anticipation or obviousness by pointing to prior art. Thus claim 27 is directed to statutory subject matter. Claims 28 and 29 depend from claim 27 and are directed to statutory subject matter at least by virtue of their dependencies.

Independent claim 31, as amended, recites "a plurality of processors P coupled to the instruction fetch catch and configured to: [all of the recited computing]." Under the broadest reasonable interpretation, the plurality of processors P must be configured or programmed in a particular manner to perform ***all*** of the recited computing. Thus, a particular machine is recited, the plurality of processors. In addition the machine imposes a meaningful limitation and is more that an insignificant extra-solution activity because ***all*** of the recited computing is performed by the plurality of processors, and thus at least one must be more than mere insignificant extra-solution activity. To the extent the Examiner contends otherwise, it is respectfully submitted that the Examiner is instead raising a rejection under Section 102 or 103, for which the Examiner is required to establish a *prima facie* case of anticipation or obviousness by pointing to prior art. Thus claim 31 is directed to statutory subject matter. Claims 32 and 33 depend from claim 31 and are directed to statutory subject matter at least by virtue of their dependencies.

Abel, Alone or in Combination with Jaber, Does Not Render the Claims Obvious

The Examiner rejected claims 1-7, 10-20 and 24-34 under 35 USC Section 103(a) as obvious over U.S. Patent No. 5,991,787 issued to Abel et al., in view of U.S. Patent No. 6,792,441 issued to Jaber. The Examiner's rejections are respectfully traversed.

The Examiner appears to rely on Jaber as allegedly teaching the distributing of the butterfly operations. The operation of Jaber for a 16-point FFT distributed among 4 processors is described below with reference to the some of the figures of Jaber, and is contrasted with the

operation of the present disclosure. This discussion is illustrative purposes only, and can be generalized to larger FFTs (size N), as well as to different numbers of processors.

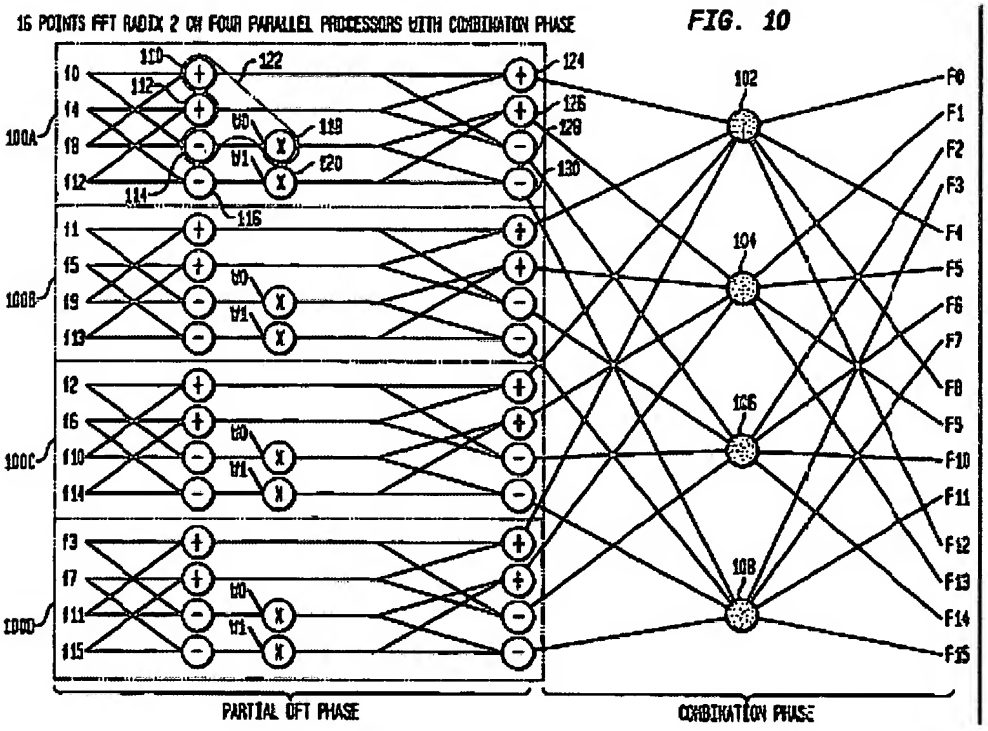


Fig 10 of US6792441 (Jaber et al)

With reference to Figure 10 of Jaber, Jaber teaches that the set of input data points (N=16) is divided among 4 processors (P=4) such that each processor works on 4 data points (i.e. executes 2 butterfly computations) in each stage of an FFT. All 4 processors execute this task in parallel without the need to send or receive data to or from the other parallel processor, for the first two stages of FFT. This is followed by a combination phase which needs the outputs (4x4=16) of all these processors and produces 16 outputs finally. The combination phase is comprised of the last 2 ($\log_2 P$) stages of the FFT.

As a result butterfly computations involving input data points are assigned to the processors as follows:

{f0, f4, f8, f12} to first processor
{f1, f5, f9, f13} to second processor
{f2, f6, f10, f14} to third processor
{f3, f7, f11, f15} to fourth processor

} G1

Until stage 2 of the FFT computation, all 4 processors execute their butterfly operations in parallel and independently of each other. Thereafter, stages 3 and 4 of the FFT computation are assigned to a combination phase. For the case of a 16-point FFT, the computations require 8 (N/2) twiddle factors/coefficients named, {W0, ..., W7}. The first two stages require the coefficients W0 and W1 only. The combination phase requires the entire set of coefficients {W0,...,W7}. This necessitates that the coefficients W0, W1 be accessible to all the 4 processors at the same time instant. This is evident from Fig 8 of Jaber, reproduced below, which shows a coefficient memory (804) being accessed by all the processors 807A,...807B.

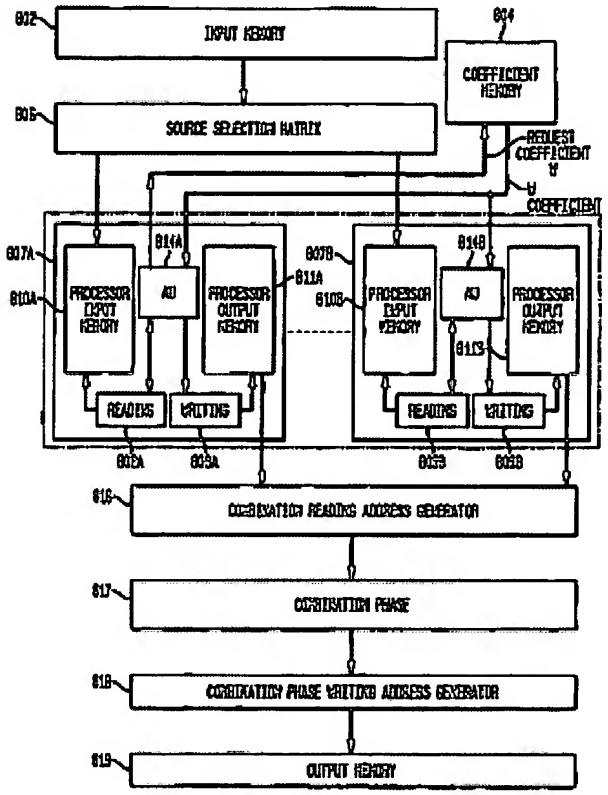


Figure 8 of Jaber

If we consider as a second example the case of a 32 point FFT (N=32) on 4 parallel processors (P=4), input data points are assigned to the processors as follows:

{f0, f4, f8, f12, f16, f20, f24, f28} to first processor
{f1, f5, f9, f13, f17, f21, f25, f29} to second processor
{f2, f6, f10, f14, f18, f22, f26, f30} to third processor
{f3, f7, f11, f15, f19, f23, f27, f31} to fourth processor

} G2

The are total 5 stages in this case. Until stage 3 of the FFT computation, all 4 processors execute their butterfly operations in parallel and independently of each other. Thereafter, stages 4 and 5 of the FFT computation are assigned to a combination phase. In this case too, the combination phase is comprised of the last $\log_2 P = 2$ stages of the 32 point FFT.

In this case there are 16 twiddle coefficients {W0,...,W15}, out of which {W0,W1,W2,W3} are required to be accessible to all the 4 processors. The combination phase requires the entire set of coefficients {W0,...,W15}.

In the general case of an N point FFT being executed on P parallel processors, there would be a total of $\log_2 N$ stages, out of which the last $\log_2 P$ stages comprise the combination phase and the remaining ($\log_2 N$- $\log_2 P$) stages are computed in P processors in parallel. In the general case, there would be N/2 twiddle coefficients in all {W0,...,$W_{N/2}$-1}, out of which the identical N/(2P) number of twiddle coefficients need to be accessible to all the P parallel processors all the while. The combination phase requires access to all the N/2 twiddle coefficients.

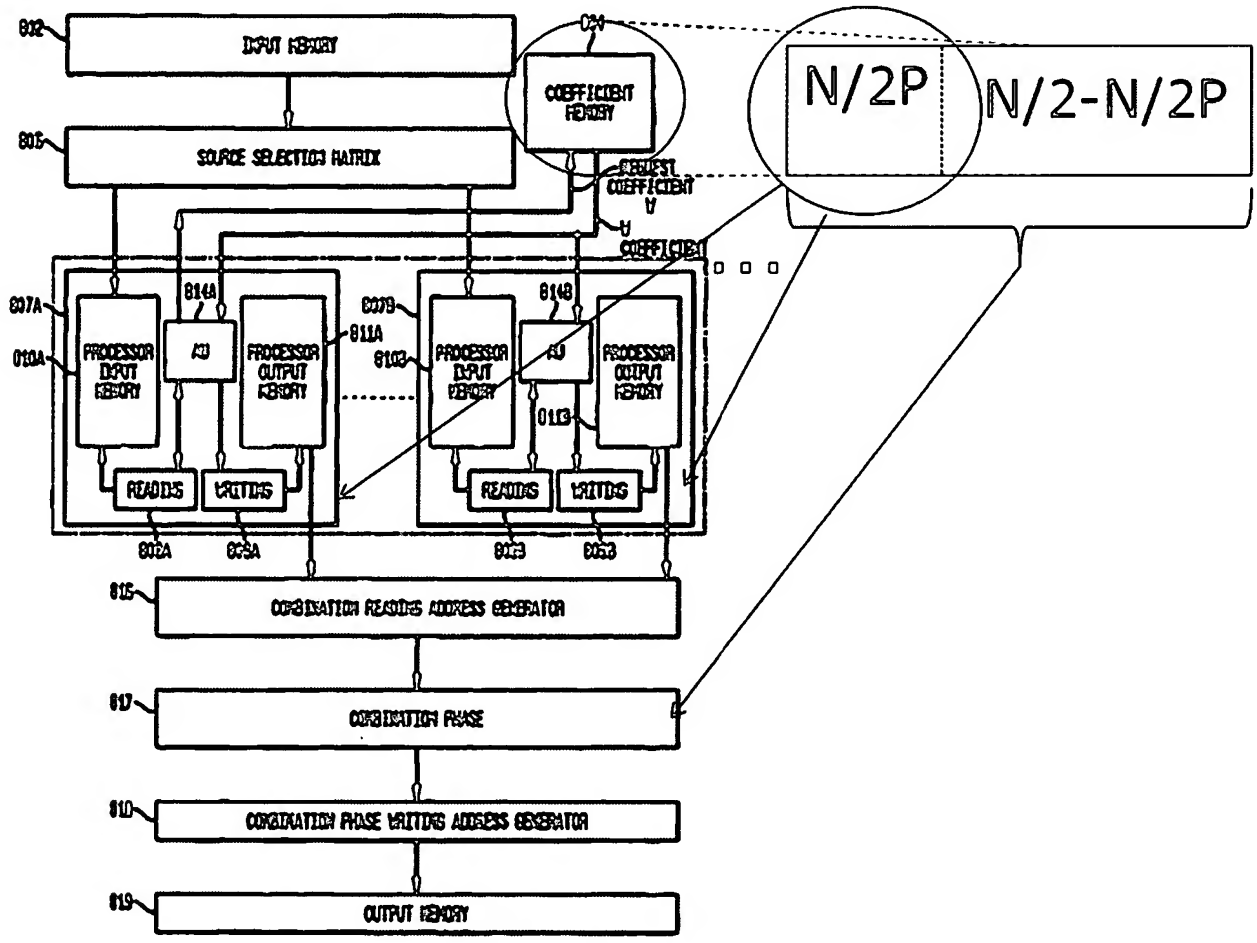
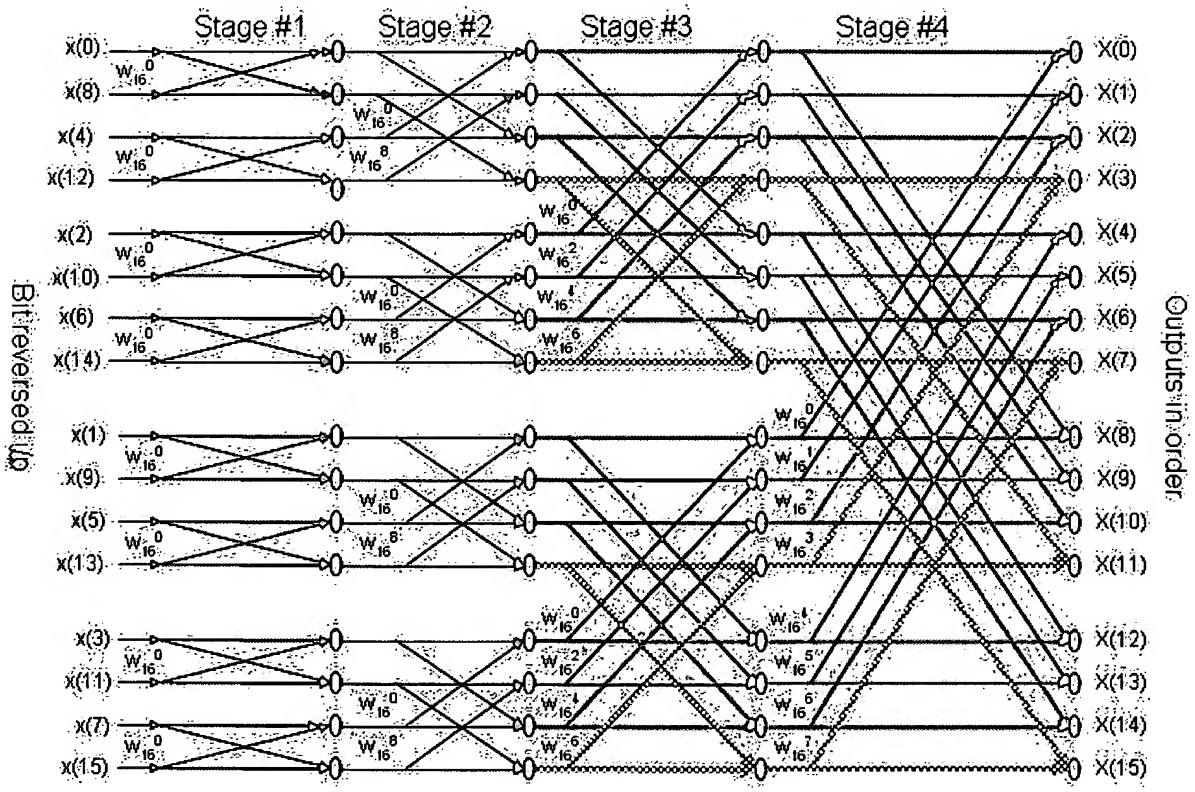


Figure 8 of Jaber Modified to Show Distribution of Twiddle Coefficients in Jaber

Figure 8 of Jaber appears again on the previous page, modified to show the distribution of the twiddle coefficients from the coefficient memory as per the Jaber. Out of N/2 coefficients, N/2P coefficients are needed by all of the processors all of the time, and the entire set is needed by the combinational phase.



Butterfly Distribution of Present Disclosure for 4-Processor Configuration: N = 16; P = 4

The butterfly distribution of the present disclosure is illustrated above for N = 16 and P = 4. The set of input data points (N=16) is divided among 4 processors (P=4) such that each processor works on 4 data points (i.e. executes 2 butterfly computations) in each stage of the FFT. (A 16 point FFT is shown to facilitate a comparison with Jaber; Figure 3 of the present application shows an 8 point FFT on a 4 processor system). All 4 processors execute this task in parallel without the need to send or receive data to or from the other parallel processors, for the last two stages of the FFT. During the first $\log_2 P = 2$, there is data dependency between the parallel processors, *i.e.* there is need of data to/from each other in the first two stages. Therefore, parallel processing of butterfly computations starts after the first 2 stages. As shown in the above figure (stage 3, stage 4), the butterflies colored in red are computed in the first processor, the

butterflies colored in blue are computed in the second processor, the butterflies colored in green are computed in the third processor, the butterflies colored in black are computed in the fourth processor. It can be seen from stage 4 that all the red colored butterflies take input data points only from red colored butterflies of the previous stage (i.e. stage 3), and so forth for the other processors. As a result butterfly computations involving input data points are assigned to the processors as follows:

{x(0),x(8),x(1),x(9)} to first processor
{x(4), x(12), x(5), x(13)} to second processor
{x(2), x(10), x(3),x (11)} to third processor
{x(6) ,x(14), x(7), x(15)} to fourth processor

} G3

After stage 2 of the FFT computation, all 4 processor execute their butterfly operations in parallel and independently of each other until the final output. Prior to that, stages 1 and 2 of the FFT computation have data dependency among the parallel processors. For the case of a 16-point FFT, the computations require 8 (=N/2) twiddle factors/coefficients $\{W^0,\ldots,W^7\}$. As shown in the butterfly distribution illustrated on the previous page, the requirement of twiddle coefficients by the 4 processors is as follows:

$\{w^0, w^4\}$ to first processor
$\{w^1, w^5\}$ to second processor
$\{w^2, w^6\}$ to third processor
$\{w^3, w^7\}$ to fourth processor

} G4

It may be noted that owing to the innovative scheme of distribution of butterfly computations among processors, the sets of twiddle coefficients required by different processors are disjoint and the number of coefficients in each set does not exceed 2.

If we consider as a second example the case of a 32 point FFT (N=32) on 4 parallel processors (P=4), input data points are assigned to the processors as follows:

{x(0),x(16),x(1),x(17), x(2),x(18),x(3),x(19)} to first processor
{x(8),x(24),x(9),x(25), x(10),x(26),x(11),x(27)} to second processor
{x(4),x(20),x(5),x(21), x(6),x(22),x(7),x(23)} to third processor
{x(12),x(28),x(13),x(29), x(14),x(30),x(15),x(31)} to fourth processor

G5

After stage 2 of the FFT computation, all 4 processor execute their butterfly operations in parallel and independently of each other until the final output. Prior to that, stages 1 and 2 of the FFT computation have data dependency among the parallel processors. For the case of a 32-point FFT, the computations require 16 (=N/2) twiddle factors/coefficients named $\{W^0,\ldots,W^{15}\}$. The requirement of twiddle coefficients by the 4 processors is as follows:

$\{w^0, w^4, w^8, w^{12}\}$ to first processor
$\{w^1, w^5, w^9, w^{13}\}$ to second processor
$\{w^2, w^6, w^{10}, w^{14}\}$ to third processor
$\{w^3, w^7, w^{11}, w^{15}\}$ to fourth processor

G6

It may be noted again that owing to the innovative scheme of distribution butterfly computations among processors, the sets of twiddle coefficients required by different processors are disjoint and the number of coefficients in each set does not exceed 4.

In the general case of an N point FFT being executed on P parallel processors, there would be a total of $\log_2 N$ stages, out of which the first $\log_2 P$ stages have data dependency between the parallel processors and the remaining ($\log_2 N$- $\log_2 P$) stages are computed in P processors in parallel without need of data to/from each other. In the general case, there would be N/2 twiddle coefficients in all $\{W^0,\ldots,W^{N/2-1}\}$, out of which disjoint sets N/(2P) of twiddle coefficients are needed by each of the P parallel processors except when P=2. In this case also

(P=2), one processor only needs all the twiddle coefficients, whereas the other one needs only a subset having N/(2P) twiddle coefficients.

The fundamental difference lies in the method of butterfly distribution among the parallel processor as shown in the assignments of butterflies among the parallel processors, G1, G3 for 16-point FFT/IFFT and G2, G5 for 32-point FFT/IFFT. As a result of this, the stages where there are dependencies among the parallel processors are in the first $\log_2 P$ stages as per the present disclosure, whereas, the dependencies among the parallel processors are in the last $\log_2 P$ stages as per Jaber.

Another manifestation of the present disclosure can be seen in the uses of twiddle coefficients in the different parallel processors. As per the present proposal, it may be noted that although all N/2 coefficients are used for the entire FFT computation, no parallel processor uses more than an N/2P subset of these, having no common coefficients with any other processors except when P=2. Whereas as per Jaber, an identical N/(2P) twiddle coefficients need to be accessible to all the P parallel processors all the while. The combination phase of Jaber requires access to all the N/2 twiddle coefficients. Figure 8 of Jaber is modified again on the following page to show how the twiddle coefficients would be applied if Jaber were modified in accordance with the disclosure of the present application. As can be seen, distinct N/2P subsets of the N/2 coefficients in the coefficient memory are used by different parallel processors. At any stage no processor has need of any coefficients which are required by any other processor. Please note that the present disclose is not limited in any way to application to the particular architecture of Jaber. It can be applicable to any system architecture in general, including shared and distributed memory systems. The use of shared coefficient memory taught in Jaber is rendered unnecessary in the present proposal due to the innovative method of distribution of butterflies among the parallel processors.

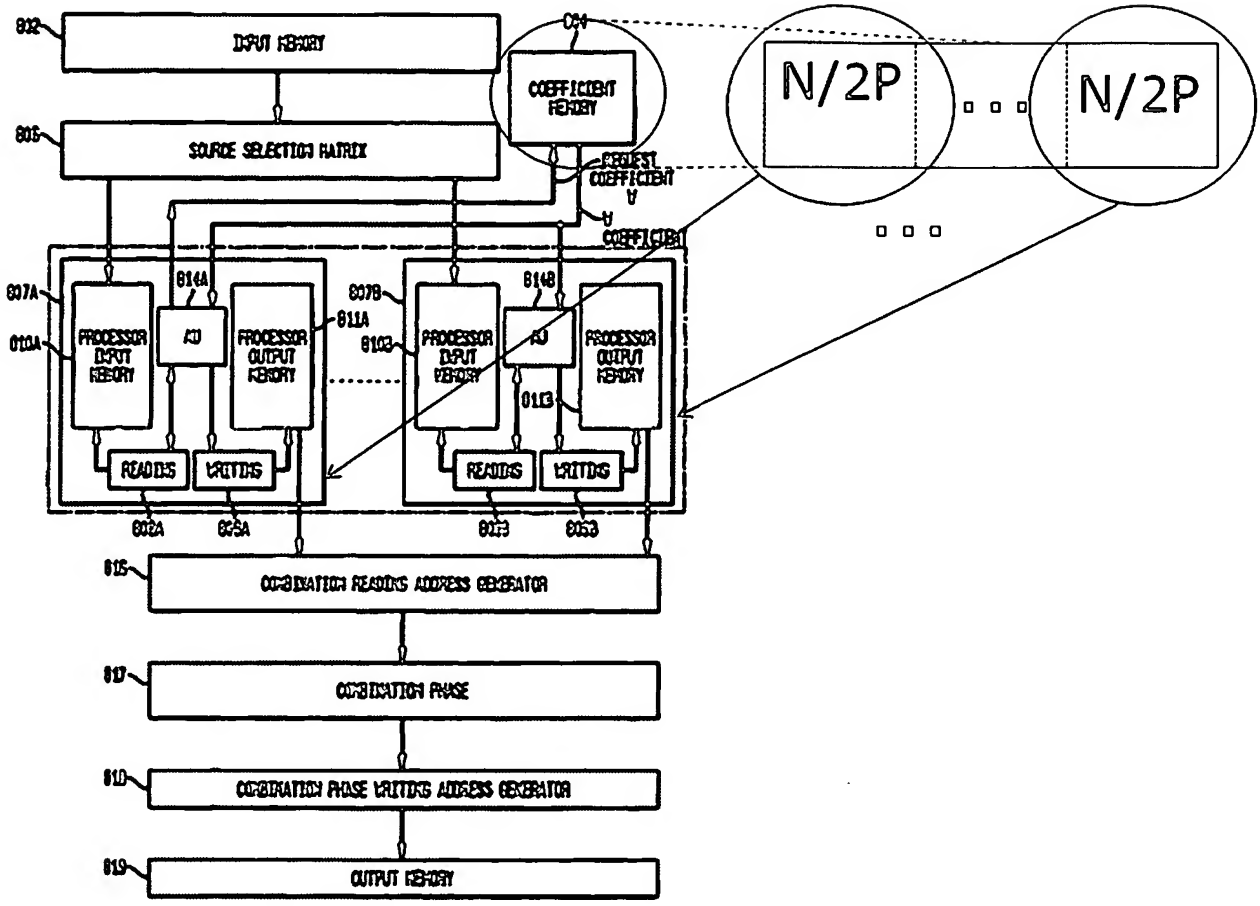


Figure 8 of Jaber Modified to Show Distribution of Twiddle Coefficients According to Present Disclosure

Turning to the language of the claims, claim 1 as amended recites, "each stage in the second set of stages has a single, un-nested computation loop; and distributing the plurality of butterfly operations in each stage of the second set of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency among the parallel processors." Abel, alone or in combination with Jaber, does not teach or suggest, or otherwise render obvious, "distributing the plurality of butterfly operations in each stage of the second set of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency among the parallel processors," as recited. Claims 2, 7 and 11 depend from claim 1 and are allowable at least by virtue of their dependencies.

Independent claim 3, as amended, recites, "means for computing a second plurality of stages of the N-point FFT/IFFT of the signal using in each stage of the second plurality of stages a plurality of butterfly operations, wherein each butterfly operation employs a single butterfly computation loop without employing nested loops; and means for distributing the butterfly operations in each stage of the second plurality of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency in the stages of the second plurality of stages." Abel, alone or in combination with Jaber, does not teach or suggest, or otherwise render obvious, "means for distributing the butterfly operations in each stage of the second plurality of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency in the stages of the second plurality of stages," as recited. Claims 4, 12, 13 and 15 depend from claim 3 and are allowable at least by virtue of their dependencies.

Independent claim 5, as amended, recites, "computing a first plurality of stages of an N-point FFT/IFFT; and computing a second plurality of stages of the N-point FFT without employing nested loops and by distributing the butterfly operations in each stage of the second plurality of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency in the stage." Abel, alone or in combination with Jaber, does not teach or suggest, or otherwise render obvious, "computing a second plurality of stages of the N-point FFT without employing nested loops and by distributing the butterfly operations in each stage of the second plurality of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency in the stage," as recited. Claim 6 depends from claim 5 and is allowable at least by virtue of its dependencies.

Independent claim 16, as amended, recites, "computing an N-point FFT/IFFT using a first plurality of butterfly computational stages and a second plurality of butterfly computational stages, each stage in the second plurality of stages employing a plurality of butterfly operations having a single, un-nested computation loop; and distributing the plurality of butterfly operations in each stage of the second plurality of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data

interdependency in the stage." Abel, alone or in combination with Jaber, does not teach, suggest or otherwise render obvious, "distributing the plurality of butterfly operations in each stage of the second plurality of stages such that each processor computes an equal number of complete butterfly operations thereby eliminating data interdependency in the stage," as recited. Claims 17 and 18 are allowable at least by virtue of their dependencies.

Independent claim 27, as amended, recites, "compute a first number of butterfly stages of an N-point Fast Fourier Transform (FFT) or Inverse Fast Fourier transform (IFFT); and compute remaining butterfly stages of the N-point FFT/IFFT with a single iterative loop wherein each processor computes an equal number of butterfly operations and there is no data dependency between butterflies in a stage of an iteration of the loop." Abel, alone or in combination with Jaber, does not teach, suggest or otherwise render obvious, "compute remaining butterfly stages of the N-point FFT/IFFT with a single iterative loop wherein each processor computes an equal number of butterfly operations and there is no data dependency between butterflies in a stage of an iteration of the loop," as recited. Claims 28 and 29 are allowable at least be virtue of their dependencies.

Independent claim 31, as amended, recites, "compute a first number of butterfly stages of an N-point Fast Fourier Transform (FFT) or Inverse Fast Fourier Transform (IFFT) of a digital signal; and compute remaining butterfly stages of the N-point FFT/IFFT with a single iterative loop wherein there is no data dependency between butterflies in a stage of an iteration of the loop." Abel, alone or in combination with Jaber, does not teach, suggest or otherwise render obvious, "compute remaining butterfly stages of the N-point FFT/IFFT with a single iterative loop wherein there is no data dependency between butterflies in a stage of an iteration of the loop," as recited. Claims 32 and 33 are allowable at least by virtue of their dependencies.

All of the claims remaining in the application are now clearly allowable. Favorable consideration and a Notice of Allowance are earnestly solicited.

Respectfully submitted,

SEED Intellectual Property Law Group PLLC

/Timothy L. Boller/
Timothy L. Boller
Registration No. 47,435

TLB:jrb
701 Fifth Avenue, Suite 5400
Seattle, Washington 98104
Phone: (206) 622-4900
Fax: (206) 682-6031

1394310_3.DOC